

## Cargadores y Ligadores

Sistemas Operativos - Semestre 2017-1

Aguilar Enriquez Paul Sebastian - 415029130

### Introducción al tema

En general, podemos distinguir dos esquemas de administración de memoria:

- Uno dice que los procesos deben estar cargados completamente en memoria para poder ejecutarse
- Uno que no requiere que el proceso completo este cargado en memoria.

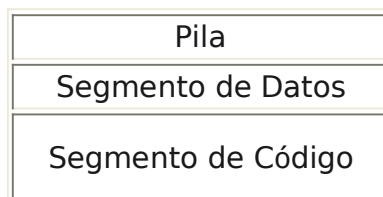
El “administrador de la memoria” es la parte del sistema operativo que se encarga de la memoria. Los sistemas de administración se pueden dividir en dos clases:

- Aquellos que sólo trabajan con la memoria principal.
- Aquellos que mueven procesos entre memoria principal y el disco, durante la ejecución.

Cuando un proceso entra en ejecución se utiliza un proceso del sistema que se denomina cargador, que lo transfiere a memoria. En Unix por ejemplo, el cargador almacena el proceso en memoria distinguiendo en él tres segmentos distintos :

- Segmento de código
- Segmento de Datos
- Segmento de Pila

Dichos segmentos se distribuyen según muestra la siguiente figura.



Al finalizar la ejecución, el área de memoria que ocupaba se declara disponible.

*“Espacio de direcciones”* es el área de memoria, desde la dirección 00000 hasta la n, en donde se ubican los procesos. Un programa de usuario puede ser ubicado en cualquier parte de la memoria, no necesariamente desde una ubicación cero.

## **Asignación de memoria**

La asignación de memoria en los segmentos es responsabilidad de diferentes etapas en el sistema.

- Compilación
- Enlazado (linker)
- Carga (loader)
- Asignación de memoria en tiempo de ejecución (run time)

## **Compilación**

Durante el proceso de compilación se generan los archivos objetos de cada uno de los archivos fuentes del programa. Las referencias externas no son resueltas en esta etapa.

## **Enlazado**

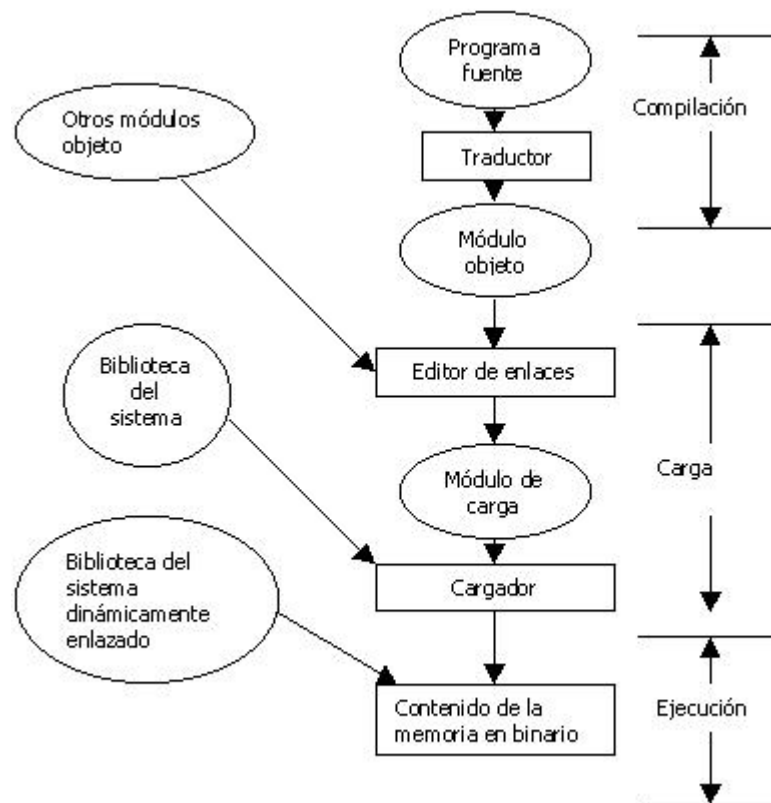
En esta etapa se reagrupan todos los archivos objetos pertenecientes al programa y se genera un solo archivo ejecutable. Las referencias externas son resueltas así como las llamadas a funciones de biblioteca del sistema.

## **Carga**

Durante este proceso se carga en memoria el programa ejecutable. Este proceso se realiza cuando se ejecuta el programa. Este carga puede permitir la “compartición” de código ejecutable entre diferentes procesos.

## **Ejecución**

Durante la ejecución el proceso se moverá de un lugar a otro de la memoria, el enlace final se deberá postergar hasta el momento de la ejecución o run-time. Para que este esquema funcione, se debe tener hardware especial.



En el caso que un programa este definido mediante un conjunto de archivos fuentes, el compilador genera el código objeto de cada uno de los archivos correspondientes. Luego es el enlazador el encargado de generar un solo archivo ejecutable a partir de todos los módulos objetos. Finalmente el cargador es el encargado de cargar en memoria el archivo ejecutable.

Un cargador es un programa del sistema que realiza la función de carga, pero muchos cargadores también incluyen relocalización y ligado.

Algunos sistemas tienen un ligador para realizar las operaciones de enlaces y un cargador separado para manejar la relocalización y la carga.

Nota\* Ambos son software base.

## CARGADORES

Un cargador es un programa que coloca en la memoria, para su ejecución, el programa guardado en algún dispositivo de almacenamiento secundario. Dependiendo de la manera en que se manejen los procesos de liga y de carga, podemos clasificar a los cargadores en:

- Cargadores iniciales
- Cargadores de arranque
- Cargadores absolutos
- Cargadores con relocalización
- Cargadores ligadores

### **Cargadores iniciales**

Indican a la computadora la forma de poner, dentro de la memoria principal, unos datos que están guardados en un periférico de memoria externa (cinta, disco, etc.). Sirven para cargar en la memoria pequeños programas que inician el funcionamiento de una computadora.

### **Cargadores de arranque (Bootstrap)**

Con la máquina vacía e inactiva, no hay necesidad de hacer relocalización de programas, tan solo se puede especificar la dirección absoluta del programa que se cargue en primer lugar. En la mayoría de los casos, este programa es el sistema operativo, que ocupa un lugar predefinido en la memoria.

**Como 1:** Una opción es que el operador introduzca en la memoria el código objeto de un cargador absoluto, utilizando los interruptores en la consola del computador.

**Como 2:** Otra posibilidad es que el programa del cargador absoluto resida permanentemente en una memoria sólo de lectura (ROM). Cuando se produce una señal de hardware, la máquina empieza a ejecutar este programa de la ROM. En algunos computadores, el programa se ejecuta directamente en la ROM; en otros, el programa se copia de la ROM a la memoria principal y se ejecuta allí.

**Como 3:** Una solución inmediata es tener una función de hardware incorporada que lea un registro de longitud fija de algún dispositivo en una localidad fija de la memoria; el dispositivo que se utilice puede seleccionarse mediante interruptores de la consola. Una vez completa la operación de lectura, el control se pasa automáticamente a la dirección de la memoria donde se almacenó el registro. Este registro contiene instrucciones de máquina que cargan el programa absoluto que sigue.

Si el proceso de carga requiere más instrucciones de las que pueden leerse en un solo registro, el primer registro causa la lectura de otros y estos, a su vez, pueden originar la lectura aún de mas registros; de ahí el término arranque. El primer registro suele denominarse cargador de arranque, y se coloca al inicio de todos los programas objeto que se van a cargar en un

sistema vacío o inactivo. Esto incluye, por ejemplo, al propio sistema operativo y a todos los programas independientes que se ejecutan sin sistema operativo.

### **Cargadores absolutos**

Como ya se menciona el programa cargador pone en memoria las instrucciones guardadas en sistemas externos. Independientemente de que sea un cargador inicial o no, si dichas instrucciones se almacenan siempre en el mismo espacio de memoria (cada vez que se ejecuta el programa cargador), se dice que es un cargador absoluto.

### **Cargadores con relocalización**

En ocasiones, un mismo programa necesita ejecutarse en diferentes posiciones de memoria. Para esto la traducción debe estar realizada en forma adecuada, es decir, no utilizando referencias absolutas a direcciones de memoria, sino referencias relativas a una dirección especial llamada dirección de relocalización.

El cálculo de las direcciones relocalizaciones lo realiza el propio cargador a medida que va guardando las instrucciones en el espacio de memoria que le indique el usuario o el propio sistema operativo de la máquina. A los cargadores que permiten estas operaciones se les denominan cargadores con relocalización.

### **Cargadores ligadores**

Montar un programa consiste en añadir al programa objeto obtenido en la traducción las rutinas externas a las que hace referencia dicho programa. Generalmente, dichas rutinas se encuentran guardadas en un fichero especial al que suele denominarse librería, porque están almacenadas todas las rutinas externas susceptibles de ser utilizadas por los diferentes programas del usuario. Allí va el programa ligador cuando esta realizando el montaje de un programa a buscarlas y las adjunta al programa objeto.

Cuando se utilizan subrutinas en un programa, el código ejecutable de cada una de ellas debe encontrarse en memoria al tiempo de ejecución. Para esto, antes de cargar un programa, debe ligarse su código objeto con los códigos objeto (guardados en uno o más archivos) de cada una de las subrutinas invocadas por él, obteniendo así un programa ejecutable que contiene tanto el código del módulo invocador como el código de los módulos invocados. En este punto, es posible guardar el resultado del proceso de liga en un archivo que podrá ser utilizado por un cargador, o el

mismo programa ligador puede también realizar la tarea de carga. Esto último evita el tener que guardar el código ejecutable en un archivo, con lo que se ahorra espacio en disco. Este ahorro de espacio en disco se paga con el tiempo gastado al tener que ligar todos los módulos cada vez que se necesite ejecutar el programa.

Este **enlace** se llama **estático** porque se realiza antes de ejecutar el programa.

Existe otro proceso llamado **enlace dinámico**, el cual consiste en enlazar en tiempo de ejecución los módulos que contienen a las subrutinas.

## **EDITOR DE ENLACE (LIGADOR)**

La diferencia fundamental entre un editor de ligado y un cargador ligador es:

Primero se ensambla o compila el programa fuente, produciendo un programa objeto (que puede contener varias secciones de control diferentes).

Un cargador ligador realiza todas las operaciones de ligado y relocalización, incluyendo búsqueda automática en bibliotecas, si se especifica, y carga el programa ligado directamente en la memoria para su ejecución.

Por otro lado, un editor de ligado produce una versión ligada del programa (llamada a menudo módulo de carga o imagen ejecutable), que se escribe en un archivo o biblioteca para su ejecución posterior.

Cuando el usuario está listo para ejecutar el programa ligado, se puede utilizar un cargador relocalizador simple para cargar el programa en la memoria.

La única modificación necesaria al código objeto es la suma de una dirección de carga real a los valores relativos del programa.

El editor de ligado realiza la relocalización de todas las secciones de control relativas al inicio del programa ligado. De esta forma, todos los elementos que necesitan modificarse en el momento de la carga tienen valores relativos al inicio del programa ligado; esto significa que el cargado se puede realizar en un paso, sin necesidad de una tabla de símbolos externos, lo que, a su vez, supone muchas menos operaciones que la utilización de un cargador ligador.

Si un programa se va a ejecutar muchas veces sin ser reensamblado, el uso del editor de ligado reduce de manera considerable la sobrecarga requerida. La resolución de referencias externas y la búsqueda en biblioteca se realizan una sola vez (cuando el programa está ligado). Por el contrario, un cargador ligador busca en bibliotecas y resuelve referencias externas cada vez que se ejecuta el programa.

### **Ligado dinámico**

El ligado dinámico ofrece algunas ventajas sobre los otros tipos de ligado. Proporciona la posibilidad de cargar las rutinas sólo cuando y si se necesitan. Si las subrutinas son grandes o tienen muchas referencias externas, se pueden conseguir ahorros considerables de tiempo y espacio de memoria.

Cuando se utiliza ligado dinámico, la asociación de una dirección real y el nombre simbólico de la rutina llamada no se hace hasta que se ejecuta la proposición llamada.

<http://www.clase.net16.net/?p=194>

- Un **cargador básico** puede cargar programas objeto en direcciones absolutas.

- Lee el registro de encabezado.
- Verifica el nombre del programa y longitud.
- Mientras lee un registro de texto lo convierte a su representación interna y lo copia a la localidad de memoria especificada.
- Salta a la dirección especificada en el registro de fin.

<http://slideplayer.es/slide/11177722/>